Universidad de los Andes
Ingeniería de Sistemas y Computación
ISIS 2203 Infraestructura Computacional

Caso de Estudio – Canales Seguros
Sistema de Gestión Empresarial y Operativa de una Compañía Transportadora
Caso 2 - Canales seguros

Objetivos

- Identificar los requerimientos de seguridad de los canales usados para transmisión de la información en el sistema de gestión empresarial y operativa de una compañía transportadora.
- Construir un prototipo a escala del sistema que permita satisfacer algunos de los requerimientos de seguridad identificados. Entendiendo las garantías de seguridad y las limitaciones de la implementación propuesta.

Problemática:

Como se indicó en el documento que describe el contexto del caso, las principales tareas del sistema son la recepción de órdenes de recogida, gestión de rutas, rastreo de unidades de distribución y paquetes, y gestión administrativa contable de recursos y de clientes.

En este contexto, surgen diferentes problemas de seguridad para algunas de las transacciones que el sistema soporta, tanto a nivel de transmisión, como en procesamiento y almacenaje de datos. Como consecuencia, es necesario evaluar riesgos y determinar medidas para mitigar los problemas detectados. Su tarea en este caso es actuar como consultor de seguridad y analizar la seguridad de las tareas relacionadas con el rastreo de unidades de distribución.

Tareas:

Suponga que la arquitectura del sistema incluye tres servidores en la oficina principal: uno se encarga del manejo y rastreo de unidades de distribución y paquetes, el segundo del manejo de órdenes de recogida, y el último se encarga del manejo administrativo y contable de recursos y clientes.

- Los puntos de atención al cliente se comunican por medio de internet con el servidor de manejo de órdenes para registrar pedidos y contratos.
- Para el rastreo de unidades de distribución y paquetes y optimización de rutas, las unidades se comunican cada 60 segundos con el servidor para informar su estado. El servidor recibe la información y la procesa. Por otro lado, el servidor de manejo de unidades de distribución calcula diariamente a la 1 a.m. las rutas del día. En condiciones excepcionales, los conductores pueden cambiar las rutas asignadas pero deben informar y justificar.
- El servidor de manejo de órdenes se comunica con el de rastreo y rutas: las rutas se calculan con base en los puntos de atención que han recibido paquetes.
- El servidor de manejo administrativo contable no atiende consultas de clientes vía web; solamente responde a consultas iniciadas en la intranet de la compañía.

A. [20%] Análisis y Entendimiento del Problema.

Suponiendo que el sistema descrito en el párrafo anterior cuenta con un firewall que filtra paquetes a la entrada de la red y antivirus en todas las máquinas de la compañía:

1. Identifique y describa los datos que deben ser protegidos en el sistema de rastreo de unidades de distribución. Explique su respuesta en cada caso (*) y responda la pregunta ¿Si un actor no autorizado consigue acceso al dato mencionado, ya sea en modo lectura o escritura, cómo podría afectar la empresa?
2. Identifique cuatro vulnerabilidades del mismo sistema, teniendo en cuenta únicamente aspectos técnicos o de procesos (no organizacionales). Identifique vulnerabilidades no solo en lo relacionado con la comunicación sino también con el almacenamiento y procesamiento de los datos. Explique su respuesta en cada caso (*).

(*) *Sus explicaciones DEBEN corresponder al contexto planteado (de forma explícita). NO se aceptarán respuestas para contextos genéricos*.

B. [10%] Propuesta de Soluciones.

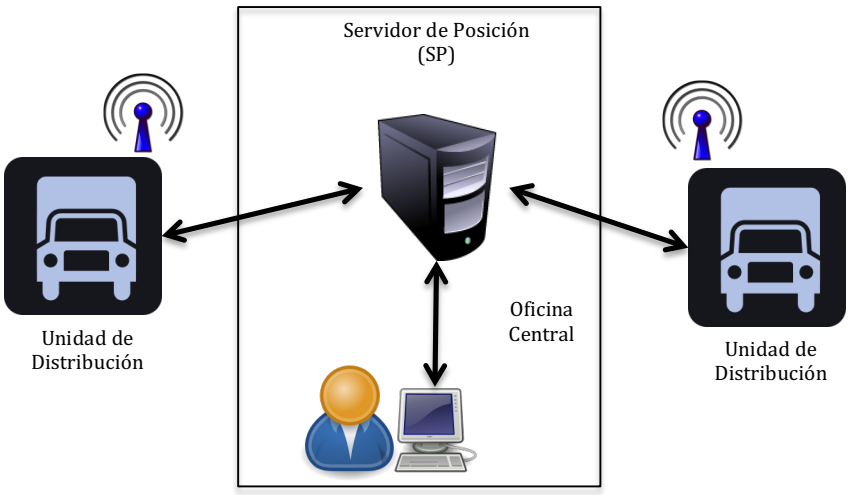
Para cada una de las vulnerabilidades que usted identificó en el punto anterior, proponga mecanismos de resolución.

- Los mecanismos propuestos deben ser explicados, por ejemplo, si se habla de cifrado sobre un canal de comunicaciones, debe identificar los participantes en la comunicación, y si es cifrado simétrico o asimétrico (y justificar la decisión).
- Además, debe justificar los mecanismos propuestos. Es decir, identifique explícitamente qué vulnerabilidad resuelve y justifique.

En sus justificaciones tenga en cuenta aspectos relacionados con eficacia, costo, eficiencia, flexibilidad, aspectos de implementación, y otros aspectos técnicos que considere convenientes.

C. [70%] Implementación del Prototipo.

En esta parte del proyecto nos centraremos únicamente en el sistema de rastreo de unidades de distribución.



Su tarea consiste en construir el cliente de una unidad de distribución que se comunique con el servidor de rastreo para reportar su estado (ubicación geográfica).

El cliente y el servidor seguirán el protocolo descrito a continuación para su comunicación:

1. El cliente se comunica con el servidor para iniciar una sesión de actualización de posición, y espera un mensaje de confirmación.
2. El cliente envía la lista de algoritmos de cifrado que usará durante la sesión y espera un mensaje del servidor confirmando que soporta los algoritmos seleccionados (si no, el servidor envía un mensaje de terminación).
3. El cliente envía su certificado digital (CD) para autenticarse con el servidor. El CD debe seguir el estándar X509.
4. El servidor verifica el certificado digital del cliente y envía su certificado digital (CD) para autenticarse con el cliente. El CD debe seguir el estándar X509.
5. El servidor genera una llave simétrica (LS) y la envía protegida al cliente.
6. El cliente recibe el mensaje y extrae la llave simétrica. A continuación usa la llave simétrica para cifrar la información de posición, y envía el código de integridad correspondiente protegido con la llave pública del servidor.
7. El servidor recibe la información y chequea integridad. Si no hay problemas la procesa localmente. Después envía respuesta al cliente, OK o ERROR, anunciado el resultado de la comunicación y la terminación.

La figura 1 ilustra el protocolo.

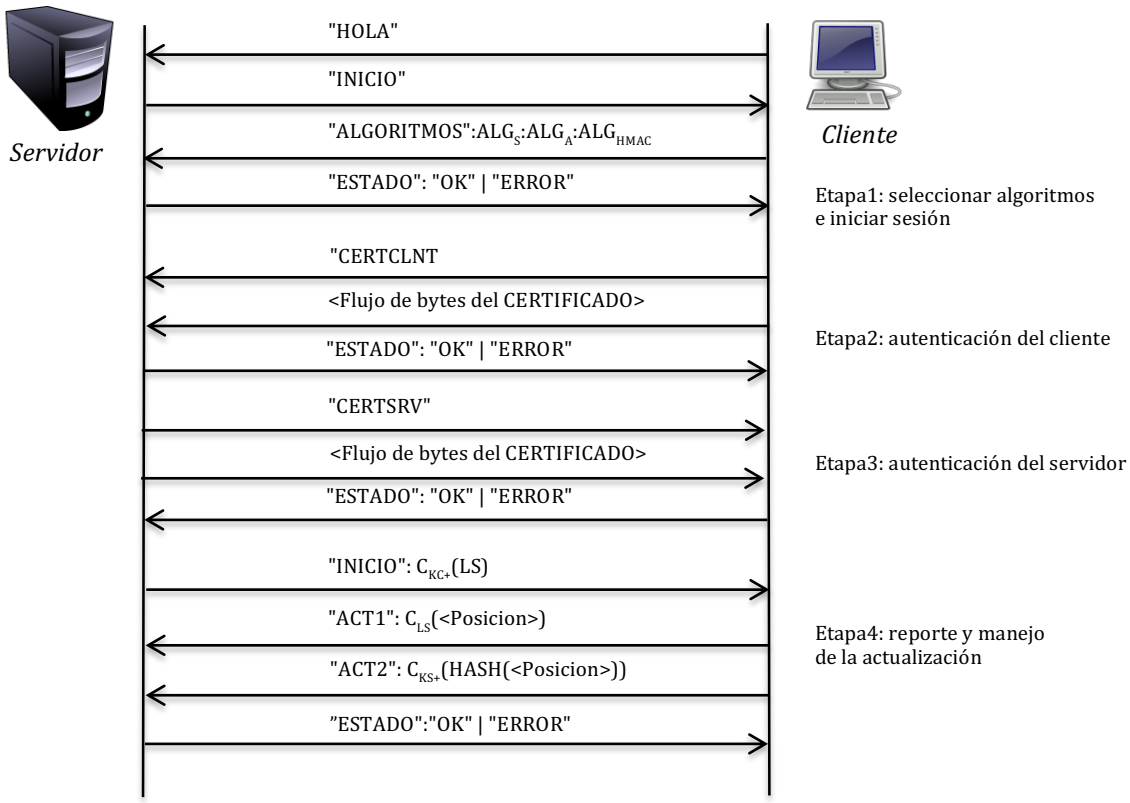


Figura 1. Protocolo de comunicación entre la unidad y el servidor.

PARA TENER EN CUENTA:

- El protocolo de comunicación maneja la siguiente convención:
  - Cadenas de Control: "HOLA", "INICIO", "ALGORITMOS", "ESTADO", "OK", "ERROR", "CERTCLNT", "CERTSRV", etc.
  - Separador Principal: " : "
- A continuación se presentan los algoritmos disponibles en el servidor para manejo de integridad y confidencialidad. Es decir, los algoritmos que deben reemplazar las cadenas ALG_S, ALG_A y ALG_{HMAC} en el protocolo. Para implementar el cliente usted debe seleccionar un algoritmo de generación de código criptográfico de hash.
  - Simétricos (ALG_S):
    - AES. Modo ECB, esquema de relleno PKCS5, llave de 128 bits.
    - Blowfish. Cifrado por bloques, llave de 128 bits.
  - Asimétricos (ALG_A):
    - RSA. Cifrado por bloques, llave de 1024 bits.
  - HMAC (ALG_{HMAC}):
    - HmacMD5
    - HmacSHA1
    - HmacSHA256

  Las cadenas que identifican cada uno de los algoritmos son: "AES", "BLOWFISH", "RSA", "HMACMD5", "HMACSHA1", "HMACSHA256".
- Utilizaremos la versión 3 del estándar X509 para los certificados digitales (CD). La idea es que el cliente y el servidor comprueben la identidad del servidor a partir de un CD (en un caso real este debería ser expedido por una entidad certificadora pero aquí se va a generar localmente). El CD debe seguir el estándar X509, en particular, debe contener la llave pública para usarla en el proceso de comunicación (se recomienda revisar la librería Bouncycastle para la generación del certificado).
- La comunicación se realiza a través de sockets de acuerdo con el protocolo de comunicación definido.
- La posición se manejará como dos parejas de números (grados y minutos en decimal), separados por una coma ",". Por ejemplo: 41 24.2028, 2 10.4418 (coordenadas usadas por Google).

- El código de envío del certificado debe lucir como se indica abajo. Es decir, primero se indica que se enviará el certificado y luego se envía el contenido (en bytes).

```
writer.println( CERTIFICADO );
java.security.cert.X509Certificate cert = certificado( );
byte[] mybyte = cert.getEncoded( );
socket.getOutputStream( ).write( mybyte );
socket.getOutputStream( ).flush( );
```

- Tenga en cuenta que en la etapa 4 del protocolo con seguridad se están enviando Strings a través de los sockets. Dado que normalmente las librerías de seguridad retornan texto cifrado como arreglos de bytes, pareciera obvio que el paso siguiente es convertir el array de bytes a un String y concatenarlo a las palabras del protocolo ACT1 o ACT2. Esto **NO** es suficiente, pues si hacen esto, el protocolo no funcionará debido al funcionamiento del encoding por defecto. Es necesario que el String que contiene la información cifrada sea codificado en hexadecimal para luego si enviarlo al servidor (Java ofrece clases para hacer esto). El servidor hará lo mismo cuando envía la llave simétrica cifrada con la llave pública del cliente. Nótese que la codificación hexadecimal debe quedar en mayúsculas para que el servidor la procese correctamente (Ejemplo: la cadena 142ae49 no es válida. La cadena 142AE49 sí lo es).
- El .jar del servidor será publicado en SICUA+. Además, se publicará otra versión sin seguridad (figura 2).

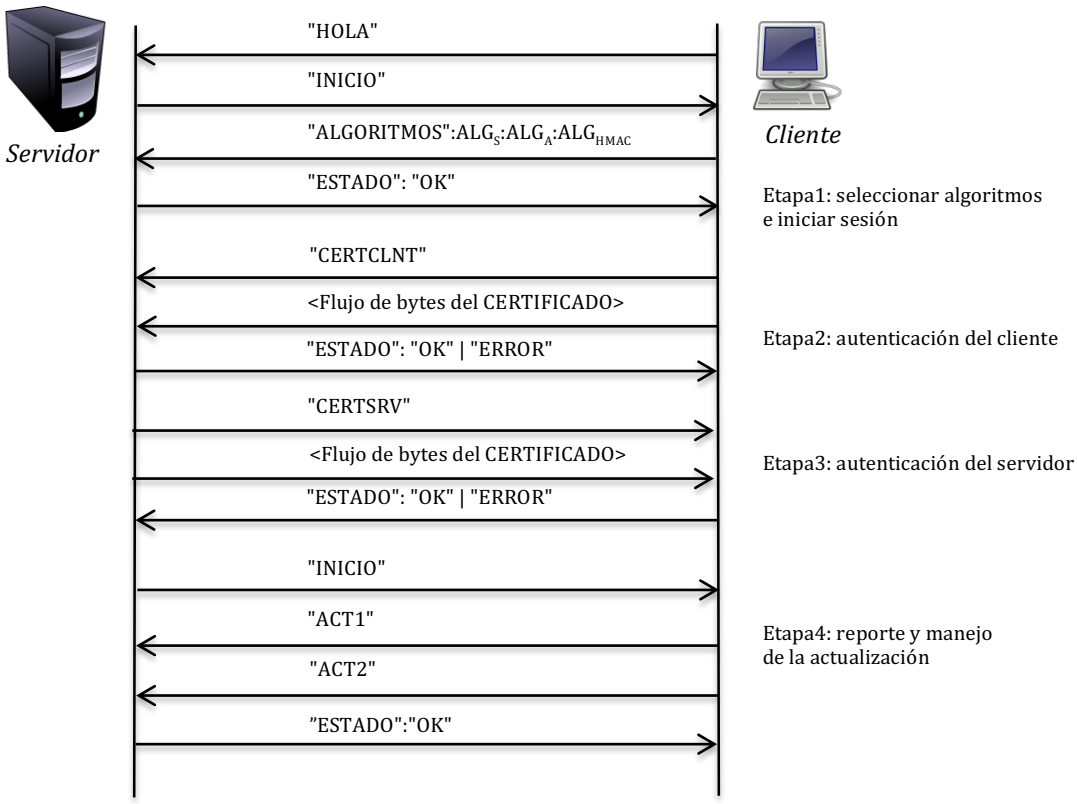


Figura 2. Protocolo sin seguridad.

Entrega:

Cada grupo debe entregar un archivo zip que incluya el informe (con las respuestas a las tareas A y B) y un proyecto Java con la implementación correspondiente al cliente (descrito en la parte C). El informe vale 30% y la implementación 70% de la calificación del caso 2.

Referencias:

- *Cryptography and network security*, W. Stallings, Ed. Prentice Hall, 2003.
- *Computer Networks*. Andrew S. Tanenbaum. Cuarta edición. Prentice Hall 2003, Caps 7, 8.
- *RSA*. Puede encontrar más información en: http://www.rsa.com/rsalabs/node.asp?id=2125

- *CD X509*. Puede encontrar la especificación en: http://tools.ietf.org/rfc/rfc5280.txt
- *MD5*. Puede encontrar la especificación en : http://www.ietf.org/rfc/rfc1321.txt